Fault Injection on FPGA implementations of Trivium Stream Cipher using Clock Attacks

F.E. Potestad-Ordóñez, C.J. Jiménez-Fernández, M. Valencia-Barrero
Instituto de Microelectrónica de Sevilla, IMSE-CNM (CSIC/Univ.Sevilla)
Email:{potestad, cjesus}@imse-cnm.csic.es, {manolov}@dte.us.es

Nowadays the security of cryptographic circuits is threatened not only by attacks on the algorithm, but also by attacks on the circuit implementation. They are the so-called side channel attacks and within such attacks are the Active Fault Analysis attacks. In literature, there are reported some vulnerability analysis of the Trivium stream cipher against Active Fault Analysis attacks using Differential Fault Analysis (DFA) [1][2]. The DFA technique is a side channel attack in which an attacker is able to inject a fault into the encryption or decryption process, thus retrieving the secret information. For the Trivium cypher, a fault is injected into the inner state. These works shown that if an attacker is able to inject only one fault in the inner state of the Trivium, the key could be retrieved, but none of them checks its feasibility on a specific hardware implementation. In this paper, it is presented an experimental analysis about the behaviour of FPGA implementations of Trivium ciphers against fault injection through the variation of the clock signal. In addition, it is made a comparative analysis between the experimental results obtained after the attack, and the expected results obtained by the simulation and timing analysis, that is, the fault positions of the Trivium inner state obtained experimentally and the fault positions expected by the timing analysis. This analysis was presented in [3] and results show the vulnerabilities of these implementations and the impossibility of determining the fault injections through simulation.

In order to achieve the fault injection over the Trivium stream cipher, it has been designed a system that generate a short pulse in the clock signal. This system was presented in [4]. Since in order to generate a short pulse in the clock signal it is necessary to know the maximum frequency at which the cipher implementation was able to work. The specific characteristics of the FPGA implementations had to be taken into account. On one hand, the FPGA uses dedicated lines for clocks, guaranteeing low delay and very low skew in the clock signals. On the other hand, the structure of the Trivium is very simple, its implementations are very fast, and work at very high frequencies. In our tests, the Trivium stream ciphers work at the maximum frequency supported by the FPGA. So the system to be developed has three main points: *a)* Frequencies well above the maximum cannot be used, because they are filtered by the FPGA. *b)* Although to insert a fault in the Trivium it is necessary a frequency slightly above the maximum using a short pulse in the clock line, the whole control system and measurement must operate at a frequency much lower than the maximum. *c)* It must be ensured that the faults are being introduced in the Trivium and it does not have any errors in the rest of the system.

To generate the short pulse on the clock signal, Digital Clock Manager (DCM) available in Xilinx FPGA devices are used. The DCM are specific modules that can generate clocks with different frequencies from an input clock. In the developed system, it has been generated a clock for the whole system and the Trivium with an optimal frequency, and another clock whose frequency is above the maximum frequency of the device. Considering that the maximum frequency of the Spartan 3E XC3S500E device is 311 MHz, a clock of 316.66 MHz has been generated because was tested that the Trivium fails at this frequency. It has been carried out several tests to verify that the clock signal of 316.66 MHz is being generated correctly. Switching between the two clock signals is done using a clock signal multiplexer, which allows switching between two clock signals without generating additional pulses.

To analyze routing dependency and behavior against the same attack, three copies of the Trivium cipher were implemented in the device in parallel mode. The short pulse was injected in two of them while the third was remained fault-free. In other words, there were two ciphers with fault injection and another that worked properly. The fault-free inner state of the third Trivium was compared with the inner states of the other two Triviums in order to detect the fault injections. To analyze the repeatability of the fault injection, approximately 1600 tests were carried out and four random pair of key and IV have been used. On the other hand, a fault injection would be considered effective or successful when a fault was injected into any of the ciphers three shift registers and this injection comprised only one wrong bit. In cases where the fault injections produced more than one wrong bit, the injection in question would be considered as an unsuccessful attack.

Table I shows the number of faults injected for each pair key-IV in the four different clock cycles. When the number of faults is one, it means that for these conditions it has been injected only one wrong bit in the shift

register and therefore the fault is effective. When the number of faults is zero means that we have not injected faults, instead, non effective faults are considered when the number of injected faults is bigger than one.

Table II and Table III show the positions of the faulty bits in the inner state for the same tests shown in Table I in the cases of key 1 IV 1 and key 2 IV 1. These results show that for the same case, the position of the fault injection can change under the same conditions, being type 1 or type 2. Even though the bits that tend to fail oscillate around the same positions, they change their number and position for the same cipher, the same pair key-IV and different insertion cycle. Regarding the bits position, it is found that the bits that tend to fail are near the bits used by the Trivium for feedback or for logic operations.

Finally, with the aim of designing an implementation capable of resisting the attacks presented, an analysis was made of delays in the bits that failed with the post-route data. The flip-flops that failed were expected to be those with the greatest delays in their input paths. Using timing restrictions on those paths would therefore reduce their vulnerability. The timing analysis with the post-route data was carried out using both static analysis tools and post-route timing simulation. Experimental results presented were compared with the results expected from the timing analysis of the implemented circuit presented in [3]. These results show that pre-implementation analyses are not valid when designing a robust implementation resistant to fault injections involving the insertion of a pulse in the clock line. The only valid analysis capable of determining the vulnerabilities of Trivium ciphers implemented on FPGA is therefore to subject the system to fault injection and identify its weakest points in experimental mode.

TABLE I. FAULT INJECTIONS ON THE CIPHERS FOR EACH KEY/IV PAIR AND INSERTION CYCLE.

	Key 1 IV 1		Key 1 IV 2		Key 2 IV 1		Key 2 IV 2	
Insertion cycles	Trivium 1	Trivium 2	Trivium 1	Trivium 2	Trivium 1	Trivium 2	Trivium 1	Trivium 2
1200	1	0	1	1	1	0	0	0
1300	2	1	1	1	1	0	0	0
1500	2-3	2	1	0	1	1	1	1
1750	1	1	1	0	1	0	1	0

TABLE II. POSITIONS OF THE FAULTS TYPE 1 AND TYPE 2 ON EACH CIPHER FOR KEY 1 IV 1.

Key 1 IV 1	Fault Type 1		Fault Type 2	
Insertion cycles	Trivium 1	Trivium 2	Trivium 1	Trivium 2
1200	2	-	-	-
1300	3/2	3	3/2	-
1500	96/3/2	-	96/3	3/2
1750	2	2	2	-

TABLE III. POSITIONS OF THE FAULTS TYPE 1 AND TYPE 2 ON EACH CIPHER FOR KEY 2 IV 1.

Key 2 IV 1	Fault Type 1		Fault Type 2	
Insertion cycles	Trivium 1	Trivium 2	Trivium 1	Trivium 2
1200	96	-	-	-
1300	3	-	-	-
1500	3/2	3	3	3
1750	2	-	-	-

From all results, it can be insured that the designed system is able to inject only one fault in the inner state of the Trivium with an efficiency of 91.34%, demonstrating the vulnerability of the Trivium against Differential Fault Analysis attacks. Furthermore, it has been seen that the bits that tend to fail are those bits used to do the logical operations of the cipher. Another point of interest is the difference in the vulnerability in relation to the key and initialization vector pair and insertion cycle. For the same key/IV pair and different clock cycles, different faults have been injected, changing them in relation to the key/IV used.

ACKNOWLEDGMENT

This work was partially supported by the Spanish Ministry of Economy and Competitiveness (with support from the European Regional Development Fund - FEDER) under contracts CITIES (TEC2010-16870), CESAR (MEC TEC2013-45523- R), LACRE (CSIC 201550E039) and MISAL (CSIC).

REFERENCES

[1] M. Hojsík and B. Rudolf, “Differential Fault Analysis of Trivium,” *Fast Software Encryption,* pp. 158-172, 2008.

[2] M.S.E. Mohamed and J. Buchmann, ”Mutant Differential Fault Analysis of Trivium MDFA”, *Information Security and Cryptology-ICISC 2014*, pp. 433-446, 2014.

[3] F.E. Potestad-Ordóñez, C.J. Jiménez-Fernández and M. Valencia-Barrero, "Experimental and Timing Analysis Comparison of FPGA Trivium Implementations and their Vulnerability to Clock Fault Injection", *Design of Circuits and Integrated Systems Conference (DCIS'16)*, 2016.

[4] F.E. Potestad-Ordóñez, C.J. Jiménez-Fernández and M. Valencia-Barrero, "Fault Attack on FPGA implementations of Trivium Stream Cipher", *IEEE International Symposium on Circuits and Systems (ISCAS'16)*, pp. 562-565, 2016.